Rhythm Engine

Search ^ K

Rhythm Engine

Quick Start

Song Setup

Samples

Audio Visualizer Sample

Looping Explosions Sample

Mania Game Sample

Mania Editor Sample

DOCS

Core

Extensions

Editor Scripts

Rhythm Engine

Rhythm Engine

Docs and Quick Start Guide

Rhythm Engine is a **high level** audio sync asset for Unity, optimized for **rapid prototyping** and **ease of use**.

Features:

- Very tight custom audio sync solution.
- Easy to prototype system based on *Scriptable Objects*.
- **Beat Sequencer** lets you quickly add rhythm-synced gameplay.
- **Audio Band Listener** lets you add stunning audio-visualizers or other audio reactive elements.
- Large collection of easy to learn, heavily commented samples, including a custom runtime editor for a Mania like game.
- Used in a real world game release - **Twin Edge**.

Created for **Unity 2021.3** and up. Tested and fully working in **Windows**, **Linux**, **Mac**, **Android** and **iOS**. It is not meant to be used with *WebGL* as it uses a different audio engine under the hood which messes up the audio sync code.

To get started, visit Quick Start Guide or try any of the Samples.

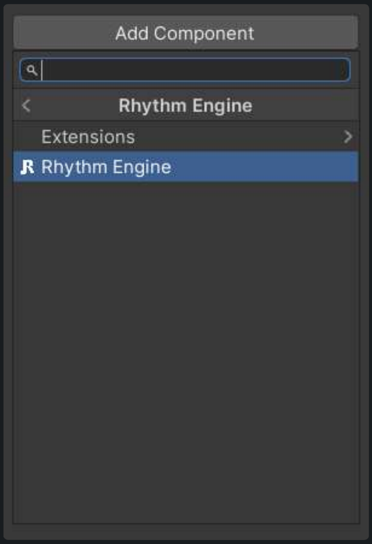
Next
Quick Start →

Last modified 2h ago

Quick Start

Getting Started

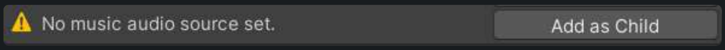
After importing the package, the first thing you want to do is to add the **Rhythm Engine** component to your scene.



Add Component → Rhythm Engine → Rhythm Engine

First choice you're going to make is choosing the **mode** that the Rhythm Engine is going to work in. **Automatic Mode** is suited for scenes which have a specific song linked to them, but if you want to change the song during the gameplay, you'd want to choose **Manual Mode**.

Both modes require a **Music Audio Source**, and because Rhythm Engine requires an Audio Source on the same GameObject, be sure to create a seperate GameObject for it.


⚠ No music audio source set.

You can also press "Add as Child" next to this warning to add it automatically.

There are also two parameters in both modes (Offset and PreStartTime), be sure to hover over them to learn more about what they do.

Automatic Mode Setup

Automatic mode requires a Song to work. Choose a demo song or create a new one by right clicking in your project folder → Create → RhythmEngine → Songs → Song/SequencedSong or by pressing the "New Song" button next to the "No Song Set." warning.

Visit Song Setup to check out how to setup a Song Scriptable Object.

With a correctly setup song, you can now press Play and everything should be up and running!

Manual Mode Setup

Manual mode doesn't require a Song to work, but can work with them. For simplicity sake, let's make sure you have a demo song prepared or create a new one by right clicking in your project folder → Create → RhythmEngine → Songs → Song/SequencedSong.

Visit Song Setup to check out how to setup a Song Scriptable Object.

With correctly setup song, you need a script that controls Rhythm Engine to play the song and start the audio sync.

Mania Editor Sample has an example on how you could make that script. But the gist of it is:

```
[SerializeField] private RhythmEngineCore RhythmEngine;
[SerializeField] private Song Song;

private void Start()
{
    RhythmEngine.SetSong(Song);
    RhythmEngine.InitTime();
    RhythmEngine.Play()
}
```

Another method of working in Manual Mode is using `RhythmEngine.SetClip()` instead of `SetSong()` which allows you to get the audio sync without using any of the premade song scripts.

Next Steps

Now you should be ready to start Rhythm Gaming! Try using the **BeatSequencer** to quickly add audio synced logic to your game or check out the **AudioBandListener** to create audio frequency based visuals! You can also use neither and use precisely timed events like in the Mania Game Sample.

Be sure to check out the Samples, they are a great resource to learn more about the asset.

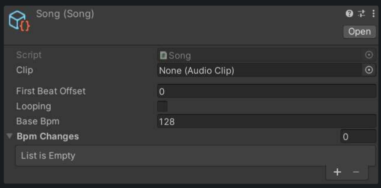
← Previous
Rhythm Engine

Next
Song Setup →

Last modified 6h ago

Song Setup

Song



Song (Song)

Open

Script: Song
Clip: None (Audio Clip)
First Beat Offset: 0
Looping: □
Base Bpm: 128
Bpm Changes: 0
List is Empty

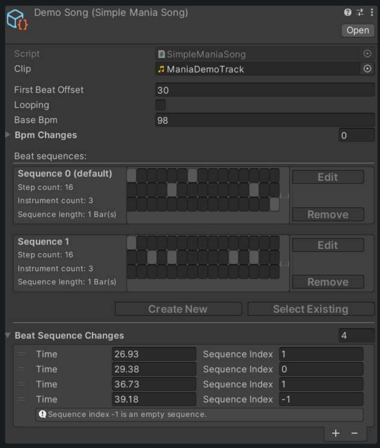
Song Inspector

The most basic song type, mainly used as a script you can extend the functionality of using inheritance.

Fields explanation:

- **First Beat Offset** - most often the song doesn't actually start on the first millisecond of the audio file or the song has a long intro, this value (in milliseconds) should be used to delay any **visuals** that need to happen. As an example in a mania game, you would use this to delay the note ***press*** time (and not to add an offset to the *input* time). Personally, I like having the editor do all the work for me, so when placing a note it automatically adds the offset time to the note time; then, in game, I don't have to worry about this offset at all, I just need to spawn notes on the time that I saved beforehand.
- **Looping** - it's important to set this flag whenever you want to loop a song, because the Engine does a tiny bit of logic under the hood to make sure the song time and audio loops smoothly.
- **Base BPM** - mainly used by the Beat Sequencer, but can be also used for example to set the distance between beat lines in a Mania Editor. You can actually completely disregard this field if you're not using BPM for anything gameplay related.
- **BPM Changes** - simple Time and BPM struct in case you ever need to change the bpm during a song.

Sequenced Song



Demo Song (Simple Mania Song)

Open

Script: SimpleManiaSong
Clip: ManiaDemoTrack
First Beat Offset: 30
Looping: □
Base Bpm: 98
Bpm Changes: 0

Beat sequences:

Sequence 0 (default)
Step count: 16
Instrument count: 3
Sequence length: 1 Bar(s)

Sequence 1
Step count: 16
Instrument count: 3
Sequence length: 1 Bar(s)

Beat Sequence Changes: 4

Time		Sequence Index	
Time	26.93	Sequence Index	1
Time	29.38	Sequence Index	0
Time	36.73	Sequence Index	1
Time	39.18	Sequence Index	-1

Sequence index -1 is an empty sequence.

Sequenced Song Inspector

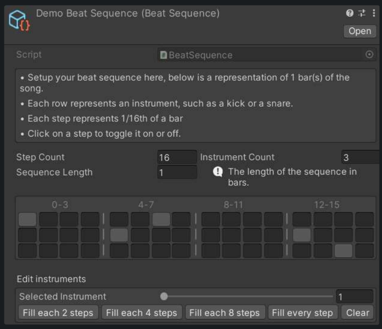
Sequenced song was made to work along with the **Auto Beat Sequencer**, it's basically just a song with additional fields to easily change the sequences during song playback.

Beat Sequencer will, by default, automatically start with Sequence of index 0.

You can't edit sequences directly in the editor of a sequenced song, but you can press the button to the right of it to force the inspector to show you the sequence in question for editing.

As you can see on the image above, if you wish to have an empty sequence playing for a duration of a song (for example during a calmer section), you can use the sequence index -1 to make the Beat Sequencer play an Empty Sequence.

Beat Sequence



Demo Beat Sequence (Beat Sequence)

Open

Script: BeatSequence

- Setup your beat sequence here, below is a representation of 1 bar(s) of the song.
- Each row represents an instrument, such as a kick or a snare.
- Each step represents 1/16th of a bar
- Click on a step to toggle it on or off.

Step Count: 16
Instrument Count: 3
Sequence Length: 1
The length of the sequence in bars.

Edit instruments

Selected Instrument: 1

Beat Sequence Inspector

Beat Sequence under the hood is, simply, a **2D Array of Bools**, where the first dimension (the X axis) represents the **beat** and the second dimension (the Y axis) represents a generic **instrument**, which can be whatever you want, such as a kick, snare, synth note, etc.

Provided custom editor should help you set up your sequence quickly.

Previous
Quick Start

Next
Samples

Last modified 6h ago

Rhythm Engine

Search ^ K

Rhythm Engine

Quick Start

Song Setup

Samples

Audio Visualizer Sample

Looping Explosions Sample

Mania Game Sample

Mania Editor Sample

DOCS

Core

Extensions

Editor Scripts

Samples

Rhythm Engine provides a couple of small samples to help you learn how to use various features of the asset. Each script has been thoroughly commented and hopefully is easy to understand by itself.

- Audio Visualizer Sample - simple demo of the **AudioBandListener** extension, features a circle surrounded by 64 lines. Each of the lines is affected by a different band in the frequency spectrum. The circle itself is controlled by the band with index 0, which usually corresponds to the kick audio frequency.
- Looping Explosions Sample - oddly satisfying demo showcasing how easy it is to add audio synced effects with Rhythm Engine. By left clicking you can place a bomb which explodes on the next snare hit in the song.
- Mania Game Sample - first part of the Mania sample, here you can play a simple 4k mania clone. It features input judgements and reacts to whatever changes you make in the attached Song Scriptable Object.
- Mania Editor Sample - second part of the Mania sample, here you can edit the mania demo Song Scriptable Object by adding and removing notes. You can freely scroll, pause and unpause the song at will to precisely control the time of the song.

← Previous
Song Setup

Next
Audio Visualizer Sample →

Last modified 5h ago